CHAPTER

8 Securing Unstructured Data

Chapter 11 discusses the subject of securing data storage—how security can be applied to the specific locations where data resides. That chapter focuses on the static state of data (information) on a hard disk or in a database, but in the high-bandwidth, mobile, and networked environments in which we work and live, information rarely stays in one place. In a matter of microseconds, information can be distributed to many locations and people around the world. In order to secure this data, we must look beyond the simple confines of *where* information can be stored and think more about *how* it is stored or, more accurately for this chapter, how it is formatted.

Information is typically categorized as being in either a structured format or an unstructured format. The meaning of these terms is subject to different interpretations by divergent groups, so first we'll address their meaning in the context of our discussion of securing unstructured data. It also makes sense to think in terms of which of three different states data is currently residing: at rest, in transit, or in use. We'll discuss that second. Finally, we'll get to the primary focus of the chapter, the different approaches to securing unstructured data.

Structured Data vs. Unstructured Data

For purposes of this book, we are not going to get into a detailed discussion about whether, for example, the unstructured Excel spreadsheet actually contains very structured data. In classic terms, *structured data* is data that conforms to some sort of strict data model and is confined by that model. The model might define a business process that controls the flow of information across a range of service-oriented architecture (SOA) systems, for example, or it might define how data is stored in an array in memory. But for most IT and security professionals, structured data is the information that lives in the database and is organized

based on the database schema and associated database rules. This means two important things to you as a security professional:

- Databases reside within a data center that is surrounded by brick walls, metal cages, network firewalls, and other security mechanisms that allow you to control access to the data.
- The data itself is structured in a manner that typically allows for easy classification of the data. For example, you can identify a specific person's medical record in a database and apply security controls accordingly.

So, because you know what structured data looks like and where it resides, you have tight control over who can access it. Security controls are relatively easy to define and apply to structured data using either the built-in features of the structure or third-party tools designed for the specific structure.

By contrast, *unstructured data* is much more difficult to manage and secure. Unstructured data can live anywhere, in any format, and on any device, and can move across any network. Consider, for example, a patient record that is extracted from the database, displayed in a web page, copied from the web page into a spreadsheet, attached to an e-mail, and then e-mailed to another location. Simply describing the variety of networks, servers, storage, applications, and other methods that were used to move the information beyond the database could take an entire chapter.

Unstructured data has no strict format. Of course, our Word documents, e-mails, and so on conform to standards that define their internal structure; however, the data contained within them has few constraints. Returning to the example of the patient record, suppose a user copies it from the web page into the spreadsheet after altering its contents, maybe removing certain fields and headers. As this information flows from one format to another, its original structure has been effectively changed.

Figure 8-1 depicts some examples of how data can move around between different locations, applications, and formats.

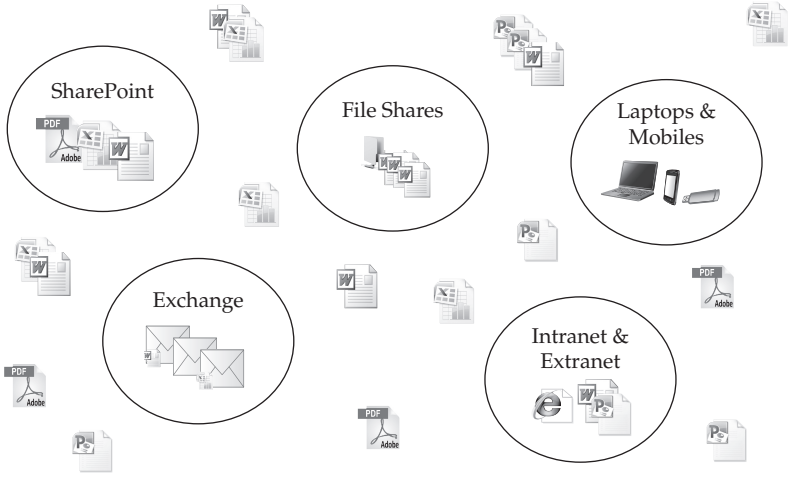

Figure 8-1 Unstructured data doesn't respect security boundaries.

Securing information when stored as structured data is relatively straightforward. But as a piece of information from the structured world moves into the unstructured world—in different file formats, across networks you didn't expect it to traverse, stored in places you can't control—you have less control. Doesn't sound good? Consider the fact that many analysts say 80 percent or more of digital information in an organization is unstructured, and that the amount of unstructured data is growing at a rate 10 to 20 times the rate of structured data. Also consider the constant stream of news articles highlighting theft of intellectual property, accidental loss of information, and malicious use of data, all with unstructured data at the core of the problem. In 2010, the worldwide total of unstructured data was estimated at roughly 1 million petabytes (1,048,576,000,000 GB) and is considered to be increasing at a rate of 25 percent a year. We clearly need to understand how we can secure unstructured data.

At Rest, in Transit, and in Use

Unstructured data can be in one of three states at any given time. It can be at rest, sitting quietly on a storage device. It can also be in transit (sometimes referred to as "in flight"), which means it is being copied from one location to another. Or, it can be in use, in which case the data is actively open in some application. Take for example a PDF file. It may be stored on a USB drive, in a state of rest. The same PDF file may be copied from the USB device, attached to an e-mail, and sent across the Internet. The PDF then moves across many states of transit as it is copied from the USB device, to the e-mail server, and travels along networks from inbox to inbox. Finally, a recipient of the e-mail actually opens the PDF, at which point the unstructured data is in use—residing in memory, under the control of an application (such as Adobe Reader), and being rendered to the user who can interact with the information.

The goal of this chapter is to focus on the challenges of securing unstructured data in all three of these states and to look at common technologies used to protect access to, and control over, information. We will look at specific types of technology and examine newer technologies, such as data loss prevention (DLP) systems, to see where the trends are going.

The Challenge of Securing Unstructured Data

To illustrate the challenge of securing unstructured data, assume your organization has an HR application that has a database which maintains information about each employee, including their annual wage, previous disciplinary action, and personal data, such as home address and Social Security number. Like most modern HR applications, it is web based, so when an authenticated user runs a report, the report is returned from the world of the structured database into the unstructured world as it is delivered to the web browser in HTML format. The user of the application can then easily copy and paste this information from the web browser into an e-mail message and forward the data onto someone else. As soon as that information is added to the body of the e-mail, it loses all structure and association with the original application.

(*continued*)

The user may also choose to copy and paste only some of the information, change some of the information, or add new content to the original information. The person to whom the user sends the e-mail may then copy and paste the information into a spreadsheet alongside other data. That spreadsheet information may be used to create a graphical representation of the information, with some of the original text used as labels on the graph. Very quickly, information can be changed, restructured, and stored in smaller data formats such as e-mails, documents, images, videos, and so on.

You might have a very well-defined security model controlling access to the HR application and the database that contains the HR information. However, the information needs to be delivered to people or applications for it to be meaningful. If it gets delivered over a network, you can make sure that access to the network is secure, yet when the information reaches the user, it can be transformed into a thousand different formats and sent to dozens of other applications and networks. Each of the locations where that information may exist can be secured, and it may be possible to apply access controls to the file share and control access to the data (content) repositories and the networks in which they reside; however, your unstructured information may end up anywhere and thus it is very hard to secure. In fact, it is hard to even locate, identify, and classify information. Once that HR data ends up deep in an e-mail thread which accidentally gets forwarded to the wrong audience, it no longer resembles the well-defined structure of the original data residing in the database. It has also been duplicated several times as it has traveled from the database to the inbox of an unauthorized user.

Unstructured data changes are constantly occurring, and data ends up in places you don't expect, particularly as the Internet provides an unbelievably large network of computers that excels in the transfer of unstructured data. Enormous amounts of money and effort have been invested in building social networking sites, file sharing and collaboration services, and peer-to-peer applications that provide endless ways in which a piece of unstructured data can be distributed within seconds to an audience of billions. It is little wonder that we frequently read about examples of data loss—now that we've created so many amazing ways to allow information to easily leave our protected borders, our network controls to stop attackers from accessing our protected data are no longer sufficient to keep it secure.

Approaches to Securing Unstructured Data

The problem of unstructured data has not gone unnoticed by the security community; we have access to an array of technologies that are designed to provide at least partial solutions to the problem of how to secure unstructured data. While what follows is by no means a complete examination of all the technologies available, we will examine the most commonly used technologies and highlight the pros and cons of each. The key areas where unstructured data can reside can be broken down into the following categories:

- Databases
- Applications
- Networks
- Computers
- Storage
- Physical world (printed documents)

The following sections describe techniques for security data in each of these locations.

Databases

The database is the center of the data world. The majority of information you are trying to secure was either created and inserted into, lives in, or has been retrieved from a database. The most secure database in the world would be one that nobody could access. It would have no keyboard attached, no network connected, and no way to remove or add storage devices. Some would even argue that the machine would also need to be powered off, located in a room without doors, and be disconnected from any power source. Clearly it would also be the world's most useless database, as nobody would be able to actually use the data stored on it. Therefore, for practical reasons, you have to secure the data within the database while also allowing legitimate users and applications to access it.

The database was once considered the realm of structured data, but with new developments in database technology, increasing amounts of unstructured data are now stored in the database. For example, the database can be the storage component of a content management system or an application that stores images, videos, and other unstructured data.

Figure 8-2 shows the typical elements of a database system. In its most basic form, the database is accessed over a network and a query is run against the database service.

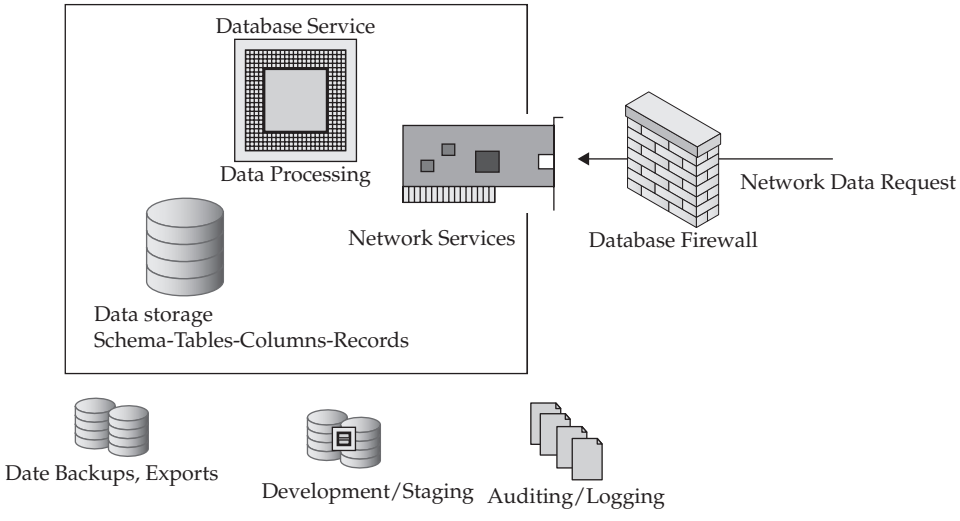

Figure 8-2 Information flows into and out of a database.

This causes a database process to run and access the data store to retrieve the queried data, which is then piped back over the network. The data store can also export data into backups that are restored on development systems or staging environments. Unstructured data can therefore reside in different areas of the database—either at rest in the schema in the database data files, in backups, or sometimes exported to other development or staging databases. Database security is discussed in further detail in Chapter 12; however, in the context of unstructured data protection, we are mainly concerned with encryption, which is discussed next.

Encrypting Unstructured Data at Rest in the Database

The most common approach to securing the data in a database is encryption (described in further detail as a general topic in Chapter 10). Encryption of data that resides in a database can be approached in various ways:

- Encryption of the actual data itself such that it is stored in normal data files in an encrypted state. The database doesn't necessarily know (or care) whether or how the data is encrypted, so it passes the encrypted data to the application to decrypt.
- Partial encryption of the database schema so that specific rows, columns, or records are encrypted as a function of the storage of the data. In this case the database handles the encryption of data and performs the decryption to the application.
- Full encryption of the database data files such that any information that resides in them is encrypted.

In the first scenario, the data itself is encrypted. As far as the database is concerned, it is just storing another big chunk of data. When dealing with unstructured data, this usually means that the files have been explicitly encrypted using some type of external technology.

The second and third scenarios are handled completely within the database platform itself—data is encrypted at rest (and also sometimes in use as it resides in the memory of the database processes). Oracle and Microsoft use the term "transparent data encryption" when referring to this sort of data security because, as far as the application accessing the data is concerned, the information is unencrypted—the encryption is transparent to the application because the data is decrypted before the application sees it. These solutions offer a variety of levels of confidentiality for the data they are protecting. Essentially the goal is to allow only database or application processes to have access to the decrypted data. Depending on the method of encryption applied, database exports and backups can be protected without additional technology. Often, data that needs to be used in development environments can be declassified by using data masking technologies that use scrambling or randomization to convert real data into fake information that still has similar characteristics to the original information.

The biggest problem with the latter two scenarios is that the unstructured data is only secured while it resides in some part of the database files. When the data needs to be accessed, it is delivered, usually in unencrypted form, to the querying application. At that point it's beyond the reach of any database encryption solution.

Implementing Controls to Restrict Access to Unstructured Data

In scenarios where the database handles encryption of data and sends the querying applications the data automatically decrypted, controlling who or what can connect to the

database and perform those queries becomes very important; this is where database access controls play a key role in restricting access to data. The approaches that are used by different databases vary, from authentication with a simple username and password to gain access to a database schema, to a complex set of rules that define for various levels of data classification who can access what, from where, at what time, and using what application. Chapter 12 discusses this subject in further detail, along with other aspects of database security. Often, as access rules become more complex, they provide opportunities and requirements to determine increasingly complex structures of data classification and control access to data at a granular level.

The credentials used to authenticate and provide authorization to access data can either be stored within the database platform or reside within an external identity directory. This enables the security of data to be associated with the enterprise directory store, thus creating an easier method for managing the access control model by using the existing access control infrastructure. For example, you may have in your identity directory a range of groups with memberships that reflect access to, say, sales, engineering, and research data. The capability to associate database permissions with such groups and have logic in the application return certain records based on those permissions provides a very effective means of controlling access. By simply moving a user from one group to another, you impact the access control mechanisms in the application through to the database.

All the investment in configuring controls to restrict access to the database still relies on trust that the process that is allowed to access the data is legitimate, or that the data continues to be secured after it leaves the database.

Securing Data Exports

Many databases provide functionality for the mass export of data into other databases. This presents security challenges. You may have encrypted the database files and, using an encrypted backup platform, tied down user access from the application through to the tables in the database, yet the owner of the schema of data may still have a range of tools at their disposal to extract and export data en mass. This activity is often the legitimate transfer of sets of data to other systems for development purposes. For example, suppose an outsourcing company is working on new features in your organization's application and requires data to work with to test its application changes. A quick e-mail or phone call by the outsourcing team to the application owner to request an export of a particular data set, and, bam, in a matter of seconds, a set of your organization's data now resides within another system that you have little to no control over.

From an unstructured data perspective, this can be a significant problem, because the information that resides in database files can be easily shipped from location to location. For example, the database may well have a set of access controls that apply only to the production instance and not other instances such as test and development. In that case, as the data is exported to test or development, you may lose those controls. That would be equivalent (in the flat-file world) to copying an entire file share containing hundreds of folders, with each folder containing confidential files, to some other file share without proper permissions. All that data suddenly becomes unprotected. Fortunately for people concerned about the security of those files, that rarely happens. But in the real world, database exports from one environment to another do occur frequently.

Encryption can be applied at the export phase. This usually is a different mechanism from that used for the encryption of data in the schema or the encryption applied to system-wide database backups. When exporting data, it is usually possible to provide a passphrase to use as the key for the one-time encryption of a specific data export. This allows sets of data to be protected in transit because the encrypted export and passphrase are shared separately when communicated to the user importing the data into their own system.

Challenges in Current Database Security Solutions

Although databases have been around since the late 1970s, they still top the list of targets for attack. And, as noted earlier, the amount of unstructured data in the database is constantly increasing. Why is this? Recently, developers have incorporated into database platforms some clever features that make the storage of unstructured data more efficient. Take for example *de-duplication*, a technique where multiple copies of the same document are automatically detected and stored only once, with a reference to the original for each copy. Consider a database that is the back-end storage for a content management system. You may have a sales presentation uploaded by many different users, and storing 20 copies of the same file would significantly increase storage requirements. By using de-duplication, you could considerably reduce your storage requirements. The database may also be able to perform data compression before the database encrypts the data at rest. So, with more unstructured data residing in the database, the database becomes a more attractive target for attack. Direct attacks to the database, indirect attacks via the application, loss of backup tapes, and poorly managed exports of sensitive data are all common threats today.

The database security methods previously mentioned are all necessary to provide a reasonable level of security while the data is resident in the database. However, at some point the data (both structured and unstructured) has to come out of the database to be presented to trusted applications. That's really the whole point of a database—not to store data forever, but to allow that data to be queried and given to other applications. Those other applications often contain their own weaknesses and are configured and managed differently, resulting in the potential for a lack of consistent data protection if those applications are not properly secured. Thus, we must follow the flow of information as it continues its journey.

Applications

Unstructured data is typically created in either of two ways: through user activity on their workstations, or as applications access and manipulate structured data and reformat it into a document, e-mail, or image. The number of applications is growing at an amazing rate. With cloud development platforms, it is now a relatively trivial task to create a collection of data from a wide variety of sources and consolidate it into a new application. For the purpose of this chapter, our focus is on web applications. There are many other types of applications, from a suite of Microsoft Office products to client/server applications that do not leverage any web-based technology. However, web applications are by far the most common network-connected application, and their client is non-specific—it's a web browser. As such, securing web applications is the greatest challenge. Secure application development is discussed in more detail in Chapters 26 and 27, and controlling the behavior of already-written applications is covered in Chapter 30.

Application security can be categorized into the following groups:

- Application access controls that ensure an identity is authenticated and authorized to view the protected data, to which that identity is authorized, via the application
- Network and session security to ensure the connection between the database, application, and user is secure
- Auditing and logging of activity to provide reporting of valid and invalid application activity
- Application code and configuration management that ensure code and changes to the application configuration are secure

This list simplifies application security into some high-level concepts that allow us to focus on the basic fundamentals of application security. Securing applications is one of the most important ways to protect data, because applications are the interface between the end user and the data. As a result a great deal of the security investment is devoted to the development of the application. Some companies, such as Microsoft, have a strict software development lifecycle (SDLC) program to ensure applications are built in a secure fashion. This approach is often called "secure by design" because security is a key part of the software development process rather than a set of configurations applied to the application as an afterthought. When you purchase an application from a vendor, these aspects of security are typically out of your control; therefore, you must choose your vendor wisely. If your organization is developing its own applications, you must ensure that your developers have a reasonable (and verifiable) level of knowledge about building secure applications. Preparing for the Certified Information Systems Security Professional (CISSP) certification exam is often a good starting point to give any developer grounding in security and the implications to application development. Chapter 27 of this book describes secure software development in further detail.

Application Access Controls

Once an application is deployed and running, the first event in the chain of events leading to access of your unstructured data is when a user attempts to gain access to the application. Most enterprise-oriented applications have the ability to integrate with existing identity management infrastructure, which is considered a best practice. This allows users to authenticate by using credentials that are already familiar to them. Single sign-on (SSO) mechanisms are typically beneficial for applications because they provide usability and security and simplify password management. Once a user has completed the authentication phase, the application must determine what the user is authorized to do by using the access control model within the application, which defines who can access which information or resources.

Network and Session Security

After the application has authenticated the user and knows which data the user is authorized to access, it must deliver that information when prompted. A wide range of protocols can be used to secure the transmission of data from the database to the application and from the application to the end user. A common protocol for securing data transmissions is Secure Sockets Layer/Transport Layer Security (SSL/TLS), which encrypts traffic delivered from the server to the client. Some authentication solutions also extend the security features of authentication to only allow certain trusted clients. This could, for instance, take the form of

allowing access to the application only by clients using a secure VPN channel or using a computer that has a certain security patch level and antivirus/anti-malware instance running. All these technologies are trying to assert a level of trust for the communication between the application and the client.

Auditing and Logging

Similar to a database, applications can also provide a variety of options for auditing and logging. This may take place in the code within the application itself, or the application web server may provide prebuilt auditing functionality that is available to the application. Again, as with a database, the audit data itself needs to be secured, as the information being recorded may also be regarded as sensitive. Security is not limited to protecting the confidentiality of audit data, though; it extends to protection of audit data from unauthorized changes—that is, ensuring data integrity. Someone who attacks a system will often attempt to hide their tracks, and being able to manipulate the audit files is often a very effective way to remove any evidence of an attack.

Application Code and Configuration Management

It is important to ensure that code and application configuration changes preserve the security controls and settings and don't introduce new vulnerabilities, a configuration management system can be used. Configuration management systems provide a repository for storing previous versions of software, as well as controls for the workflow of reviews and approvals needed to enforce compliance with an organization's security policies.

Many large applications can also be configured to work with governance, risk management, and compliance (GRC) software that has fine-grained knowledge of the application assets as well as the policies the organization wishes to comply with, providing reporting that verifies compliance with policy. A good example of a GRC solution in action is ensuring that a doctor who is permitted to prescribe drugs doesn't also have the ability to dispense them (separation of duties). GRC solutions are embedded tightly with applications and can also work alongside identity management solutions.

Challenges in Current Application Security Solutions

Applications are among the biggest generators of unstructured content, and the aforementioned security mechanisms highlight the need for a sufficient level of security to ensure that only authorized users are able to access an application, pursuant to the principle of least privilege. Securing this trusted connection ensures that content is delivered to the end user confidentially.

Application security is the most mature and robust area of data security, because it's been around longer and has received more attention than other areas such as networks and computers. The big application vendors who create the sort of software that runs many large corporations have invested significant time and effort to secure these applications. Large consulting firms have been built on their expertise in deploying these applications in secure configurations, and the applications are surrounded by technologies that secure the communication of information into and out of the application. Yet in the same manner that a database has to move information from its secured world and give it to a trusted application, the application must secure access to, and the transmission of, data to the end user. A user may well be using the correct authentication credentials and accessing from a secure network, and may only be given data that is relevant to their role, but ultimately they

are getting access to information that, with today's browser-based web applications, they can very easily remove from the secure confines of the application into the totally unknown and unprotected world of unstructured content.

This is why organizations continue to experience data loss. Security may be implemented at the application level, yet once the information goes beyond the application, there is little or no control over it. As soon as the data is copied beyond your controlled network, you lose the ability to protect the information.

Networks

Data moves from the protected realm of the database into the application and on to the end user. Sometimes this communication occurs via a local process, but often the application and database reside on different servers connected via a network. The end users are rarely on the same computer that the application and server reside on. Therefore, the security of the network itself is another area that we must examine. However, because several other chapters of this book focus on the subject of network security, the scope of this section is limited to mentioning some of the technologies designed to secure unstructured data on the network. These technologies are discussed in more depth in other chapters.

Network security technologies have developed into complex systems that are able to analyze traffic and detect threats. Network intrusion prevention systems (NIPSs; see Chapter 18) actively monitor the network for malicious activity and, upon detection, prevent intrusion into the network. Malware protection technologies prevent Trojans from deploying and planting back doors on your trusted network clients. The newest polymorphic advanced persistent threats (APTs), which steal data, provide back doors to attackers, and cause denial of service (DoS) attempts, can be blocked using solutions that detect illegitimate traffic and deny it.

All of the network security techniques described in Part IV of this book can be used to secure network communications. However, network security solutions are not able to protect information that has already left the network; they can only detect the unauthorized transmission of data and deny it from going any further.

Challenges in Current Network Security Solutions

Again, as with application security for unstructured data, the biggest challenge with network security for unstructured data is that once a trusted user or client is connected to the network, the network will freely pass to that user or client any information they are authorized to access. You might be using advanced techniques to monitor traffic flows, but to enable business, you still need to allow information to flow. For example, suppose you need to secure information in collaboration with a new business partner. Suppose further that you've created a secure network between the two businesses and integrated it with the identity management systems to only authenticate legitimate users at the partner business to access your networked applications. Over this network will flow many pieces of valuable information, securely. However, when that information reaches the partner, it will no longer be bound within your nicely secured network and may exist in the clear, meaning it can be easily lost or misused. If something goes wrong, such as loss of data, you may have nothing but audit records showing who from the partner business accessed that data. Those may be useful in figuring out what went wrong, but any measures you take in response will always be reactive, after the damage has already been done.

Two growing trends in the modern business environment are to move to cloud-based services and to expand external collaboration in the form of outsourcing, partner alliances, and increased customer access to company information. This has resulted in the creation of various methods for connecting your corporate network to several sources, further complicating the problem of security. Data flows across networks to the application and from the application over the network to the database. Network security solutions should be used to provide the best possible protection to this data.

Computers

Once a legitimate user has securely connected across the network to the application to access data residing in the database, the information is ultimately presented in a web page that is rendered by a web browser. From there, the user can move the data to an unstructured and unprotected location, such as a PDF file or an Excel spreadsheet, and then download and store the data on the local drive of a desktop workstation. Therefore, the security on the computer from which the user interacts with the application and resulting unstructured content becomes critical. Essentially, the computer is the front line in the battleground of information security today.

Servers are usually limited in number and physically under your control, or at least under the control of your cloud services provider. Networks and their gateways are also limited in number and usually within your control. But end-user computers may number in the hundreds or thousands and may often be beyond your security control. Furthermore, those computers may be running a large number of platforms, a wide variety of OS versions, and a wide variety of software, and may be used by a wide variety of people. Everything we do with information happens on the computer, which means the security of that computer is critical to the ongoing security of your information. Within the context of the security of unstructured content on computers, we will focus on only a few areas:

- Ensuring that only legitimate users can access the computer (identity access control)
- Controlling the flow of information over network interfaces and other information connection points (USB, DVD, etc.)
- Securing data residing at rest on the computer

Identity Access Control

All operating systems have some form of user access control. The most basic (and common) form is a username and password combination that is validated against a local identity store. Successfully authenticated users are then granted a session to the system, which then manages access to resources, typically by using an access control list (ACL). In business environments, computers under the control of the organization are configured to authenticate and grant sessions to users whose identities are stored in a centralized identity management system. The most prevalent identity management system is Microsoft Active Directory, which natively manages access to Windows clients but is also often extended to the management of Unix-based workstations. (Access control is covered in further detail in Chapter 7.)

It is this identity that is associated with any unstructured data on the system. Unfortunately, once a piece of unstructured data resides on a computer, any security mechanism that controls access to it on the system is longer effective. Thus, the identity that is used in the computer access may have little effect on the actual access controls to local content. Some technologies are able to shrink the access control layer to the content itself, and that is the subject of the following chapter.

Controlling the Flow of Data over Networks and Connected Devices

Once a user is authenticated and provided with a session, they are able to proceed to access the network, read data on the local hard disk, and connect USB and other storage devices and transfer data to and from the computer. Privileges to perform these actions are implemented by the local computer, which in turn, with technologies like Active Directory, conforms to a central policy that defines what users are able to do on their desktops.

From the unstructured data perspective, all these connections are simply pipes through which information can travel, from one to the next. Each pipe may have its own security model, but each model applies security to the unstructured data only while it travels through the associated pipe.

Securing Data at Rest

Once they are logged in to a computer, most users in organizational environments start working with data that is often stored locally on the host device. It might, for example, take the form of a Word document saved from an e-mail to the desktop; or it might be a PDF file cached by your browser as it displays it from a web page. Even smartphones are computers that are able to store massive amounts of data and run business software that allows the use of unstructured data. Security for storage devices, as described in Chapter 11, becomes important at this stage.

Challenges in Current Computer Security Solutions

Computers represent the interaction of humans and computers. Much effort goes into creating a level of trust between the person and the computer. Usernames, passwords, security token keys, smart cards, fingerprint scanners, various biometric devices, and other mechanisms all attempt to ensure the computer knows that the person is who they claim to be. Yet once this trust has been established, information is allowed to flow freely. This trust in the human at the keyboard often fails. Humans send e-mails by accident, lose USB keys, print confidential information and leave it lying around, and sometimes even steal data intentionally.

Consistently securing information in use on computers is one of the biggest challenges facing the information security community today. Yet the computer is the perfect place to start educating the end user on the importance of security. There are mature solutions for different aspects of computer security. Anti-malware technologies have been on the market for over a decade, and storage encryption is in a fairly well-defined space. The challenge is to integrate the various endpoint technologies to accurately support your security policies.

Storage (Local, Removable, or Networked)

Once on the computer, information either exists in a dynamic state, in the memory of a running process (for example, with a web browser or software application), or is stored in an unstructured form on the hard disk or a removable drive (for example, in a file located in a local folder).

Storage is one of the most effective areas of unstructured data security and is often the one which receives the most attention after a data loss incident. Storage security solutions mainly deal with data at rest, but sometimes stored data exists in another state, either in transit or in use. An example of this is the contents of a PDF file that is stored in RAM and paged to the disk by the operating system. Is this data at rest or in use? (An interesting court case that hinged on the distinction between "at rest" and "in use" was mentioned in Chapter 3.) Typically, storage security solutions focus on encryption or access control. These subjects are covered in more detail in Chapter 7, but let's consider the strengths and weaknesses of encryption (detailed in Chapter 10) further before we move on.

Encryption of Storage

The most common "go to" security tactic for anyone who has suffered a data breach, failed an audit, or just wants to be proactive is to "encrypt everything" because that seems to be the easiest and most comprehensive approach. This is mainly because databases, network storage, content management systems, and computers ultimately end up storing data on storage devices like hard disk arrays or USB flash memory, so encrypting these locations is an obvious solution. Methods of encryption on storage devices fall into two categories:

- Disk encryption (either hardware based or software based)
- File-system encryption

Disk Encryption Hardware disk encryption is totally transparent to the operating system and, therefore, to applications and users on the computer. This means storing a piece of unstructured data on the disk is simple and requires no change to the operating system, the application, or the content format.

Some form of authentication must take place before any data on the drive can be decrypted and read. When full disk encryption is performed using software, the operating system must have some form of unencrypted partition from which to boot and in turn authenticate the system to gain access to the cryptographic keys to decrypt the main encrypted disks.

Both hardware and software methods provide the encryption of data as it is written to disk across the entire disk. The methods are totally transparent to the processes (other than, of course, those in the operating system that may be managing the encryption/decryption), and as data is read from disk, it is seen by applications in its decrypted state. One downside of hardware-based disk encryption is that the management of the cryptographic keys can be difficult.

File-System Encryption Another method of encrypting data at rest is to implement the functionality in the file system itself. This means that the methods may vary depending on the operating system in use. Typically, the main difference is that file and folder encryption only secures files and folders, not the metadata. In other words, an unauthorized person is able to list the files, view the filenames, and see the user and group ownerships, but they cannot actually access the files themselves without the cryptographic keys. (An exception is the ZFS file system, which does encrypt metadata.)

File-system encryption, like disk encryption, only applies when the content resides in a location that is encrypted. If the data is moved from the encrypted disk to a disk that's unencrypted, it is no longer protected.

Challenges in Current Storage Security Solutions

Storage encryption and access controls are common in many organizations, yet occurrences of data loss continue to increase, even from environments that have these types of solutions. As mentioned, encryption only works at the source.

Storage access controls are clearly necessary. You need a level of control over users' access to information on local computers or networked file shares. However, the limitations of these access controls are very quickly reached. Imagine the simple scenario of two folders on a network file share, "All Company - Public" and "Confidential - Engineering." You may have defined a tight set of engineering groups who have access to the latter, while allowing anyone in the company, maybe even outsiders, to access the former. It only takes the single act of a trusted engineer to accidentally store a few confidential documents in the "All Company - Public" folder to render the access controls for that information completely ineffective. Location-based access controls like this only provide control for the location where the information resides; information moves everywhere, but the security defined at the location does not travel with it.

Data Printed into the Physical World

A lot of time and effort is devoted to finding solutions for securing unstructured data in the digital world; however, data loss incidents due to the loss or inappropriate disposal of paper documents must also be considered—that is, finding solutions for how to secure information in the hardcopy world. Considering data in paper form as unstructured content is also another way to visualize how structured data can easily pass into the unstructured world. You print it out.

Encryption typically doesn't help when printing information because the information needs to be readable to humans. Instead, methods such as watermarks and redaction are used. Watermarks leave identifying data (like a background image or word) in the printed copies in an attempt to alert users to the importance of the information and try to reduce the chances they are negligent with the data. Redaction is the process of editing or blacking out certain text in a document such that certain sensitive parts of a document are not visible. Redaction often applies to the visibility of both the digital document and the resulting printed version. Risks are introduced when information is not redacted sufficiently, or the wrong data is redacted. For example, the U.S. Transportation Security Administration (TSA) in 2009 posted a PDF online that had been redacted in an ineffective way such that the blacked-out areas could be easily viewed simply by copying the content and pasting it into another document.

The computer industry has been promising the paperless office for many years, and although we collectively generate a great deal less paper than we used to, paper has definitely not gone away. Many data breach incidents are the result of poor disposal of physical paper records.

Physically printed copies of documents can pose risks just as great as those posed by their digital counterparts. A user could print a spreadsheet and send it via fax or mail to an unauthorized party. No information security technology that exists in the digital domain is ever going to detect, secure, and control access to the information when it is

printed on paper. You can identify every single place that digital information is going to live and encrypt those locations, but that encryption has no effect on a paper-based copy. Thus, the best practices for handling printed documents are limited to restriction of the contents printed, along with reliance on human vigilance.

Before being printed, all confidential documents should have any non-essential contents hidden or deleted, so unnecessary confidential information is not included in the printout.

All printed confidential documents should have a front cover page, so their contents are not visible from a casual glance. Also ensure that all pages have page numbers (so any missing pages can be detected) and watermarks along with headers and footers identifying the confidentiality level of the document.

Each copy of an entire document should be labeled, to aid in tracking each copy. And confidential printouts should never be taken home, to a restaurant or any public place, or anywhere outside of the controlled environment they're meant to be in.

Confidential documents should be printed to a private printer, whenever possible, instead of to shared printers located in common areas. When this is not possible, the person who prints the material should go directly to the printer, watch it print, and collect the output immediately. Confidential documents should never be left unattended, even for a short time.

All paper documents containing confidential information should be locked in a secured container such as a desk drawer or file cabinet. They should never be left sitting on top of a desk, even for a short time.

When no longer needed, documents should be immediately shredded or placed in a secure container for a shredding service to destroy. The quality of shredding is important—older "strip shredders" that simply cut pages into individual strips are surprisingly easy to put back together. Crosscut shredders cut in more than one direction, resulting in diamond-shaped pieces that are much more difficult to reconstruct.

The *Handbook for Safeguarding Sensitive Personally Identifiable Information* published by the U.S. Department of Homeland Security provides some further guidelines for secure handling of hardcopy:

- When faxing data, make the recipient aware that a fax is about to be sent, so the recipient will be aware that they need to go collect the fax right away.
- When sending mail, verify that the correct recipient received the delivery. Also make sure the envelope is opaque so the contents inside are not visible. Use tracking information to see where the delivery goes, via a delivery service that provides Return Receipt, Certified or Registered mail, or a tracking service.

Finally, and as always, common sense should be used by the people who handle paper copies of documents, because so much of document security is dependent on human behaviors. People handing out confidential documents need to make sure they know who they are giving copies to, and keep track of those copies. They should also be responsible for collecting and destroying those copies when done. A good example is a handout used in a meeting as a reference. After the meeting is concluded, the meeting organizer should make sure to take back the handouts, and remind everyone about their confidentiality. How do you enforce good behaviors like this? The best way is to ingrain the desired behaviors in the overall culture of the environment, through consistent messaging and repetition via a security awareness program (as was discussed further in Chapter 5).

Newer Approaches to Securing Unstructured Data

The previous sections of this chapter described various techniques for securing unstructured data in individual environments, which may be thought of as "point solutions." These techniques emerged from security requirements of individual use cases, and each is focused on the capabilities and limitations of the environment to which it pertains. Newer approaches to unstructured data security are broader in scope, more data-centric, and less platform-dependent. The following sections describe these newer approaches, and how they can be used to complement the security capabilities mentioned above.

Data Loss Prevention (DLP)

Data loss prevention (DLP) refers to a relatively new group of technologies designed to monitor, discover, and protect data. You might also hear this technology referred to as data *leak* prevention—and sometimes it's also referred to with the word *protection* instead of *prevention*. In any case, DLP is like a "firewall for your data." There is a wide variety of DLP solutions on the market, which typically can be broken down into three types:

- **Network DLP** Usually a network appliance that acts as a gateway between major network perimeters (most commonly between your corporate network and the Internet). Network DLP monitors traffic that passes through the gateway in an attempt to detect sensitive data and do something about it, typically block it from leaving the network.
- **Storage DLP** Software running either on an appliance or directly on the file server, performing the same functions as network DLP. Storage DLP scans storage systems looking for sensitive data. When found, it can delete it, move it to quarantine, or simply notify an administrator.
- **Endpoint DLP** Software running on endpoint systems that monitors operating system activity and applications, watching memory and network traffic to detect inappropriate use of sensitive information.

Network, storage, and endpoint DLP are often used together as part of a comprehensive DLP solution to meet some or all of the following objectives:

- **Monitoring** Passive monitoring and reporting of network traffic and other information communication channels such as file copies to attached storage
- **Discovery** Scanning local or remote data storage and classifying information in data repositories or on endpoints
- **Capture** Storage of reconstructed network sessions for later analysis and classification/policy refinement
- **Prevention/blocking** Prevention of data transfers based on information from the monitoring and discovery components, either by interrupting a network session or by interacting with a computer via local agents to stop the flow of information

DLP solutions may comprise a mixture of the above, and almost all DLP solutions leverage some form of centralized server where policies are configured to define what data should be protected and how.

Challenges in DLP Solutions

If the DLP solution isn't monitoring a particular storage device or network segment, or if a particular file doesn't have the right policy associated with it, then the DLP solution cannot enforce the right level of protection. This means that every network segment, file server, content management system, and backup system must be covered by a component of the DLP technology along with proper classification of all documents critical to its success. Proper configuration of the DLP environment and policies is a big task, and overlooking one (or more) aspects can undermine the whole system.

DLP only makes point-in-time decisions. Consider, for example, a DLP policy that allows users to send confidential data to a trusted partner. Six months later the organization decides this partner is too expensive, and sets up an agreement with another, cheaper partner. You then reconfigure the DLP policy to reflect this change in the business relationship, but the DLP solution has no ability to affect all the information that has flowed to the old partner. Data may now reside with a partner who will soon be signing an agreement with your competitor.

The way in which a DLP solution deals with policy violations has limitations. Prevention is part of its name, and for good reason—when a user is copying a file or e-mail, the DLP solution prevents copying of information that it deems illegitimate. This may initially seem like a good outcome, but what if your CEO is giving an important presentation and wants to copy a file to an unencrypted USB stick to share a marketing presentation with the board of directors? DLP might block that. What if you want to e-mail an important document to your home e-mail address to work on over the weekend? Nope, DLP blocks it and informs the IT security group. While DLP has many advantages, it may impact business processes and productivity if all possible scenarios are not considered.

DLP is also capable of generating a certain number of false positives (and false negatives), which makes fully implementing all blocking/prevention components a risky exercise. Even when the accuracy of policy enforcement is very high, organizations often find the disruption to business so high that they prefer a monitoring-only implementation.

Despite these problems, DLP is still an excellent tool for hunting down and monitoring the movement of sensitive data. It can provide very valuable insight into the information flows within the network and, at a minimum, can highlight where illegitimate activity takes place or where sensitive information is stored in open file shares. The reports from DLP network monitoring and discovery components provide a useful feedback loop: identifying compliance "hot spots" and poor working practices, mapping the proliferation of sensitive content throughout (and beyond) your enterprise, and enabling organizations to tune their existing access control systems. But keep in mind that you will need to add additional trained staff to reap all of the benefits of the DLP solution. A significant increase in workload is required to examine and act on all the alerts coming from the DLP solution, including false alerts.

Information Rights Management (IRM)

Information rights management (IRM) is a relatively new technology that builds protection directly into the data files, regardless of where they are stored and where they are transmitted and used. IRM evolved from digital rights management (DRM), which is used in the entertainment industry to protect music and movies and apply protection to all kinds of data.

IRM uses a combination of encryption and access controls to allow authorized users to open files, and to block unauthorized users. With IRM, files are encrypted using strong encryption techniques. When a request is made to open the file and decrypt the data, software is required to check with a central authentication server (usually somewhere on the Internet) and, via a reliable handshake mechanism, determine whether the requesting user is allowed to unlock the data.

IRM solutions go even further than providing access to the data; they control the ability to copy, paste, modify, forward, print, or perform any other function that a typical end user would want to perform. This provides a granular level of control over files that can't be provided by any other single security technology. Thus, IRM can be a valuable tool in the security toolbox. Continue to the next chapter to learn much more about this promising technology.

Summary

As defined in this chapter, unstructured data is any collection of electronic information that does not follow a strict format (and therefore lacks any inherent security controls). Unstructured data by itself is wide open and unprotected. This data may reside in databases, applications, networks, computers, storage, and even the physical world. And, it is growing at a pace faster than that of structured data, which typically has inherent security controls.

Structured data is relatively easy to secure compared to unstructured data, which exists in three states at various times: at rest, in transit, and in use. Unstructured data is found in applications, networks, computers, storage systems, and even inside structured databases. And once that data is printed into the physical world, it can no longer be controlled by the software-based security technologies applied to the original source data. This chapter provided an overview of a range of security technologies for securing unstructured data in all of these locations, including the limitations of those technologies.

With regard to applications, we discussed the use of access controls, network and session security, auditing and logging, and configuration management to protect unstructured data at rest. For the network, we identified that you can use all of the network security technologies described in Part IV of this book to protect data in transit. On computers, we covered ensuring that only authorized users can access the data, controlling the flow of that data, and securing the data at rest on the computer. On storage systems, we mentioned access controls and discussed encryption in more detail. We looked at securing unstructured data within databases through the use of encryption. In the physical world, we reviewed the challenges and best practices for handling paper copies of confidential information. And finally, we talked about two newer technology solutions: data loss prevention (DLP), and information rights management (IRM) as solutions to protecting unstructured data.

There is a common thread to these technologies and what they attempt to achieve: ensuring that only authenticated and authorized users can access secured data, and that bypassing the access control protecting that data isn't easy. These technologies also share a common challenge: they need to be implemented in the proper places, and even when they are, if the information travels beyond your perimeter of control, you lose control over that information and visibility of where it goes thereafter. As with most security models, layering of various security controls helps to close the gap by providing a defense-in-depth approach to security.

References

Birru, Amha. *Secure Web Based Voting System for the Case of Addis Ababa City: Securing Vote Data at Poll Stations, In the Wire and Data at Rest.* VDM Verlag, 2009.

ICON Group International. *The 2011–2016 Outlook for Information Data Loss Prevention (DLP) Appliances in the United States.* ICON Group International, 2011.

Kenan, Kevin. *Cryptography in the Database: The Last Line of Defense.* Addison-Wesley, 2005.

National Institute of Standards and Technology. *NIST Special Publication 800-111: Guide to Storage Encryption Technologies for End User Devices.* NIST, 2007. http://csrc.nist.gov/publications/nistpubs/800-111/SP800-111.pdf

Photopoulos, Constantine. *Managing Catastrophic Loss of Sensitive Data.* Syngress, 2008.

U.S. Department of Homeland Security. *Handbook for Safeguarding Sensitive Personally Identifiable Information.* http://www.dhs.gov/xlibrary/assets/privacy/privacy_guide_spii_handbook.pdf.

U.S. Government. *Guide to Storage Encryption Technologies for End User Devices.* Books, LLC, 2011.